On Autistic Interpretations of Occam's Razor:

Explanatory and Creative Alternatives to the MDL principle

José Hernández-Orallo

Universitat Politècnica de València
Departament de Sistemes Informàtics i Computació,
Camí de Vera 14, Aptat. 22.012 E-46071, València, Spain
E-mail: jorallo@dsic.upv.es.

Ismael García-Varea

Universitat Politècnica de València
Institut Tecnològic d'Informàtica,
Camí de Vera 14, Aptat. 22.012 E-46071, València, Spain
E-mail: ivarea@iti.upv.es.

Abstract. The Minimum Description Length (MDL) principle is the modern formalisation of Occam's razor. It has been extensively and successfully used in machine learning (ML), especially for noisy and long sources of data. However, the MDL principle presents some paradoxes and inconveniences. After discussing these all, we address two of the most relevant: lack of explanation and lack of creativity. We present new alternatives to address these problems. The first one, intensional complexity, avoids extensional parts in a description, so distributing compression ratio in a more even way than the MDL principle. The second one, information gain, forces that the hypothesis is informative (or computationally hard to discover) wrt. the evidence, so giving a formal definition of what is to discover.

Keywords: MDL Principle, Model Evaluation, Scientific and Knowledge Discovery, Occam's Razor, Intensional Complexity, Machine Learning, Explanatory Induction, Informativeness, Creativity.

1. Introduction

The maxim "induction as compression" has become increasingly popular in Predictive Modelling and Machine Learning since R. J. Solomonoff recognised (Solomonoff 1964) that the unsupervised learning of a grammar from raw data may be understood as information compression.

Some other relevant landmarks in this trend have been the Minimum Message Length (MML) principle (Wallace & Boulton 1968), the view of pattern recognition as data compression (Watanabe 1972) up to Rissanen's milestone of the Minimum Description Length (MDL) principle (Rissanen 1978).

They all are "fresh interpretations" (Conklin & Witten 1994) under Algorithmic Information or Stochastic Complexity of a much older idea attributed to William of Ockham 1290?-1349?:

Occam's Razor Principle:

"If there are alternative explanations for a phenomenon, then, *all other things being equal*, we should select the simplest one."

The MDL principle represents a formal and sound incarnation of Occam's Razor. Since its appearance in 1978, the principle has been extensively used in practice (see e.g. Quinlan & Rivest 1989, Cheeseman 1990, Derthick 1990, Muggleton et al. 1992, Zemel 1993, Pfahringer 1994, Conklin & Witten 1994) and the ML literature is full of assertions like: the shorter the theory the better, the more likely, the more plausible...

This has motivated some overhype about the MDL principle, which has motivated radical claims like "all kinds of computing and formal reasoning may usefully be understood as information compression by pattern matching, unification and search" (Wolff 1995).

Although we share the view that the MDL philosophy is positive, we think that the MDL principle should only be used justifiably in the prototypical case of large and *non-random* data from noisy sources. However, the MDL principle presents many problems for the inference to the best explanation (loosely known as abduction) and creative (or informative) induction, two cases of non-deductive inference which are essential for scientific discovery and everyday reasoning.

In this paper we highlight these and other problems of the MDL principle and we present two alternatives which ensure that the hypotheses are explanatory and creative.

2. Formalisation and Usage of the MDL principle

Occam's razor has been frequently rejected because there *was* no objective criterion for simplicity, Popper being the major partisan of this position. However, Stochastic Complexity and Kolmogorov Complexity are well-established criteria of simplicity. Moreover, Algorithmic Complexity $C(\cdot)$ or Kolmogorov Complexity $K(\cdot)$ (see e.g. Li & Vitányi 1997) is been gradually recognised as a key issue in statistics, computer science, artificial intelligence, epistemology and cognitive science.

DEFINITION 2.1. KOLMOGOROV COMPLEXITY

The *Kolmogorov Complexity* (KC) of a string x given y on a descriptional mechanism (or bias) β :

$$K_\beta(x|y) = \min \{ \ l_\beta(p_x(y)) \ \}$$

where p_x denotes any "prefix-free" β-program for x using input y and $l_\beta(p_x)$ denotes the length of p_x in β.

The complexity of an object x is denoted by $K_\beta(x) = K_\beta(x|\varepsilon)$ where ε denotes the empty string. It can be seen elsewhere (e.g. Li & Vitányi 1997) that Kolmogorov Complexity is an absolute and objective criterion of simplicity, and it is independent (upto a constant term) of the descriptional mechanism β.

In absence of any other knowledge about the hypotheses distribution, one choice is the prior distribution $P(h) = 2^{-K(h)}$. This is precisely what R.J.Solomonoff proposed as a 'perfect' theory of induction. This prior was popularised by J. Rissanen in 1978 as a general modelling method, under the name of the popular MDL principle:

Minimum Description Length Principle (Rissanen 1978):

"*The best model to explain a set of data is the one which minimises the sum of: the length, in bits, of the description of the theory; and, the length, in bits, of data when encoded with the help of the theory. Then, we enclose the exceptions, if any*".

This two-part code formulation (the hypothesis + the data encoded) has recently been modified to a one-part code (Rissanen 1996), which is almost exactly the same as the *ideal* MDL principle (Vitányi & Li 1996), i.e., the best description y for some data x is the one such that $l(y) = K(x)$.

Since the principle is not computable in general, it is usually approximated or used in restricted descriptive mechanisms, like attribute languages. The main motivation of its success is that it avoids overgeneralisation (overfitting) when the data is noisy or it may contain errors.

In (Vitanyi and Li 1997) and (Li and Vitanyi 1997) it is shown that in many cases Bayesian reasoning "is prone to overfitting" while MDL is not. Referring to the "ideal MDL principle", Vitányi and Li discuss (Vitanyi and Li 1996) that "*with a more complex description of the hypothesis H, it may fit the data better and therefore decreases the misclassified data. If H describes all the data, then it does not allow for measuring errors. A simpler description of H may be penalized by increasing the number of misclassified data. If H is a trivial hypothesis that contains nothing, then all the data are described literally and there is no generalization. The rationale of the method is that a balance in between seems required [...]*". This is the reason why the MDL principle has become popular as a means to avoid over-generalisation (underfitting) and under-generalisation (overfitting).

Theoretically, the MDL principle is closely related to the Minimum Message Length (MML) principle and Maximum Likelihood Estimators (Case & Smith 1983) . It has also been compared with cross-validation (Kearns et al. 1999) and Bayesian Learning (Gull 1988).

Philosophically, the MDL principle matches with Kuhn's notion of "changing paradigms" (Kuhn 1970): *Exceptions are patched until they are long enough to force the revision of the paradigm (or model) of the theory.*

3. Some Problems of the MDL principle.

Our discussion is motivated from the apparent contradiction between the so-called "no-free-lunch" theorems about induction (Wolpert 1992, Schaffer 1994) which state that one learner cannot be better than another when performance is averaged uniformly over all possible problems. These results only allow that a learner could be better than another for a particular distribution of problems.

Vitányi & Li (Vitányi & Li 1997) show that the MDL principle is almost optimal for the universal distribution $2^{-K(x)}$. Of course, the universal distribution (i.e. Occam's Razor formalised) is just a choice when you have no information at all about the real origin of the information.

This choice, although successful in many applications, presents many theoretical and practical problems. Most problems have been eclipsed because the MDL principle is a reasonable option for almost every kind of learning paradigm. In the end, there is an important methodological reason: if the model is not much predictable, at least it is short and manageable.

1. The first problem is that $K(\cdot)$ is not computable. At first sight, it seems that the goal of MDL modelling is to obtain shorter and shorter theories for a given data x, and it is not relevant to know where is the limit $K(x)$. However, the prior $P(h) = 2^{-K(h)}$ is not computable, and the posterior probabilities must be approximated. Even using a computable approximation, like Lenin's variant $Kt(\cdot)$ or simply the length of the hypothesis, the principle turns out to be ultimately *relative*, because the probability of a hypothesis dynamically changes as the learner knows that something can be further compressed.
2. *Frequently unmanageable*: For the sake of maximum compression, the theory can be computationally intractable. This problem can be solved in a similar way as the previous problem, by using the Levin's variant which compensates length and computational time.
3. *Perfect data*: the MDL under-fits perfect data: new examples are quoted until their compression is worthy. In the words from Vitányi and Li: "*with some amount of overstatement one can say that if one obtains perfect data for a true hypothesis, then Ideal MDL interprets these data as data obtained from a simpler hypothesis subject to measuring errors. Consequently, in this case Ideal MDL is going to give you the false simple hypothesis and not the complex true hypothesis.*" (Vitányi & Li 1996).
4. *Short data*: all the theoretical justifications of the MDL principle and its relationship with other principles are asymptotic, i.e. when the size of the data

grows to infinite. This has been clearly recognised by Grünwald (Grünwald, 1999) "(T)*he MDL Model Selection Criterion (...) will only work well if one really has a lot of data, since the neglected constant can be very large*.". It is important to realise that, for short data, the MDL hypothesis is usually the extensional data itself.

5. *Discontinuous*: The reliability of the theory is not always increasing with the number of examples that have confirmed the theory. E.g. the sequence $(a^n b^n)^*$ is more compressible if $n=10^{10}$ than if $n=78450607356$.
6. *Inconsistent with Deduction*: the use of the MDL principle to obtain a hypothesis is usually separated from any deductive work with the hypothesis, because any deduction could increase the length of knowledge. For instance, given T_a and T_b, intuition (and logic) says that $T = T_a \vee T_b$ should have more probability, but the MDL principle assigns less probability to T because it is larger. Is is remarkable to see that ILP (Muggleton & De Raedt 1994) has successfully used variants of the MDL principle because the representational language, Prolog, does not allow disjunction in the head.
7. *Frequently non-explanatory*: Under the MDL principle paradigm, for the sake of maximum mean compression, some parts of the data are left as exceptions. Consequently, these have no predictive character. Recently, the initial MDL principle two-part code formulation has been corrected to a one-part code (Rissanen 1996). It has been argued that this correction solves the problem of partially extensional descriptions, but this is absolutely false. The same problem can appear intrinsically, some part can be very compressed (the main rule) and other parts are quoted as exceptions.
8. *Frequently Non-Informative or Non-Creative*: Popper advocated for informative hypotheses, because they are more falsifiable. Moreover, it is important to distinguish the hypothesis generation from their evaluation. An inductive algorithm or learner can generate one or multiple hypotheses. Its performance should be measured by the informativeness and creativity of the hypotheses it generates. An algorithm that generates almost always evident hypotheses from the data is not very useful for scientific discovery and many other learning paradigms. The worst case takes place when the data x is random, namely the extremely frequent case $K(x) \approx l(x)$, and the hypothesis is simply the data extensionally quoted. Apart from the problems of explanation, because the data itself does not explain anything, it provides null information.

Most of these problems are produced because the MDL principle is autistic: "*the principle simply obtains the hypothesis which is suggested by the data*" (Vitanyi & Li 1996). In other words, there is no goal for explanation, no idea of surprise or interest in the result of the learning algorithm. Are we always so autistic about the source of the information that we pretend to discover? Is this the case in many applications of machine learning, scientific discovering or even cognition?

In the following we will present two alternatives to the MDL principle for explanation (7) and discovery (8). Aimlessly, problems (1) to (4) are also solved by the first alternative and (6) is clarified by the second. A completely different approach to solve these problems (including problem 5) is presented in (Hernandez-Orallo 1999).

4. Explanation and Intensionality

In the previous section we have identified a major problem of the MDL principle for explanation, some part of the data is left as an extensional quotation. It is understood as noise, and this means that it is not comprehended. This extensional part is not validated, making the whole theory weak. An ontology is difficult to construct from here if the exceptions are unrelated (not explained) with the other facts.

Although intensionality (avoidance of extensionality) and explanation are very close notions, there is an important difference between them. An explanation must be manageable in order to communicate it and convince oneself and others. We will first undertake the notion of intensionality, extensively used in mathematics and philosophy but never formalised, and then we will give an explanatory adaptation of it.

4.1. Intensional Descriptions

The distinction between extensional and intensional description of a set or a function is something that is learnt in elementary school. It is soon realised that infinite sets can only be described by intensional descriptions. Intensionality is also closely related to the classical and fundamental philosophical problem of sense, reference and meaning, but there has not been presented to date any definite account or framework to distinguish intensional and extensional definitions for *finite* data and *general* description languages.

For instance, nowadays, there is no general and objective way to answer to the question: what is the difference between the description "the numbers 2, 4, 8, 16, 32, 64" and the description "the powers of 2 with less than 3 digits". One of the best ideas to approach this problem is to think that an intensional description is a 'compressed' description. This would recognise the MDL principle as the best principle to obtain intensionality. The next example shows that this is not the case for finite data:

EXAMPLE 4.1. Consider the following finite data: "1, 2, 3, 5, 7, 11, 13". From the infinite many possible descriptions we show some of them:
Extensional Description:

D_1 = "1,2,3,5,7,11,13"

Partially Intensional Descriptions:

D_{2a} = "Odd numbers until n=13 with positive exception 2 and negative exception 9".

D_{2b} = "The 9–2 first odd numbers with positive exception 2 and negative exception 9".

Intensional Descriptions:

D_3 = "The values e_1 to e_7 given by the following series: $e_{-1}=0$, $e_0=0$ and $e_i= e_{i-2}+e_{i-1} + 1-\{(i-1) \text{ div } 2\}$".

D_{4a} = "Prime numbers until n=13".

D_{4b} = "The first 7 prime numbers".

D_{4c} = "Prime numbers until n<14".

D_5 = "The roots of the polynomial P" where P is easily computed to satisfy that $P(x)= 0$ iff $x \in \{1,2,3,5,7,11,13\}$.

Intensional? Description:

D_6 = "The roots of the polynomial P_1 and the roots of the polynomial P_2" where P_1 is easily computed to satisfy that $P_1(x)= 0$ iff $x \in \{1,2,3,5\}$ and P_2 is easily computed to satisfy that $P_2(x)= 0$ iff $x \in \{7,11,13\}$.

The first description, D_1, would probably be the shorter for most of descriptional languages, because the other require a lot of auxiliary concepts, which make them larger. This illustrates that the MDL principle is not useful for short data. D_{2a} is a partially intensional description because part of the data (2 of 7 examples) are given extensionally, as exceptions. Finally, D_3, D_{4c}, and D_5 are apparently fully intensional descriptions.

The way to obtain fully intensional descriptions is to avoid extensionalities, i.e., exceptions, as the following subsection formalises.

4.2. Formalising Intensionality

Example 4.1. shows that the idea of exception as "recognising something in the description exactly equal to the data" is easy to cheat. Some part of the description can be entangled to make difficult the comparison (see e.g. D_{2b}). Conversely, some part of the description may casually match with the data (see e.g. D_{4a} and D_{4b}). Moreover, descriptions D_3 and D_5 show that an exception is something that could be 'disguised' in many ways, even a systematic way (D_5). In any case, the only way to avoid these 'disguises' must be based on the comparison of the mean compression ratio instead of a direct comparison of the information.

Given a descriptional language or machine ϕ, we will denote with $\phi(T)$ the extension (or output) of a theory or program T.

DEFINITION 4.1. COMPRESSION RATIO

The Compression Ratio of a theory T wrt. an evidence E is defined as:

$$CR_E(T) = l(E \cap \phi(T)) \,/\, \{\, l(T) + l(\phi(T) - E) \,\}$$

If E is omitted then $CR(T) = CR_{\phi(T)}(T) = l(\phi(T)) \,/\, l(T)$, which is the usual formula of compression ratio.

From here, an exception could be approximated by any subtheory or subprogram E of a theory T such that $CR(E) < CR(T)$. However, there are auxiliary parts which do not cover anything *alone* and, in this way, almost any theory would have exceptions. Nonetheless, this idea could be used conversely, by detecting a *general rule* G which is more compressed than the whole theory.

DEFINITION 4.2. COMPENSATED DESCRIPTION

A theory or description T is non-compensated iff there does not exist a proper subtheory or subprogram G of a theory T such that:

$$CR_{\phi(T)}(G) \geq CR(T)$$

This definition accounts for all the descriptions in example 4.1. For instance, descriptions D_3 to D_5 are compensated because there is not a subprogram with more compression ratio. By the use of the subscript $\phi(T)$, the trick of selecting the subprogram "prime numbers until 1000" of D_{4a} is not valid because $l(\phi(G) - \phi(T))$ is great. In contrast, D_{2a} and D_{2b}, would not be compensated if the definition of odd is short because the ratio of the subdescription "odd numbers until n=13" will be high because $l(\phi(G) - \phi(T))$ is low (only there is a negative exception 9). On the other hand, if the description of odd is long, then the description "positive exception 2" has better compression ratio than D_{2a}. The limit case is the fully extensional description D_1, where any part can be chosen as a subdescription with the same compression ratio as the whole description. Finally, D_6 is not compensated.

Definition 4.2 represents the idea of compensated compression. The comparison $\geq$ is used instead of $>$ in order to make the definition more independent of the description mechanism (if, casually, it turns to be that $CR_{\phi(T)}(G) = CR(T)$, there are many other languages where this equality would not hold).

In general, the idea of intensionality must also avoid *extensional* exceptions, i.e., parts whose compression ratio would be less than 1. The formalisation of this is more difficult because there are many ways to select a part that covers few evidence (or none). Maybe the best idea would be to extend the notion of general rule into the following one:

DEFINITION 4.3. PATCHED GENERAL RULE

A theory or subprogram G is a *patched general rule* of a theory T iff:

$$CR_{\phi(T)}(G) \geq CR(T) \text{ and } l(\phi(T) - \phi(G)) \,/\, \{mf_q \cdot (l(T) - l(G)) + af_q\,\} \leq 1$$

where af_q and mf_q are the additive and multiplicative factors of quoting. For most representational languages $mf_q = 1$ and af_q is very close to 0 and it can be discarded (the length of introducing some instruction or code like "PRINT x").

A refinement of definition 4.2 by using this extension is direct:

DEFINITION 4.4. INTENSIONAL DESCRIPTION

A description T is intensional iff it has no *patched general rule*.

The term $l(\phi(T) - \phi(G)) / \{l(T) - l(G)\}$ represents the compression ratio of the rest (the exception). Obviously, any compensated description is also an intensional description.

Under this definition 4.4, D_6 is still not intensional. However if we have the description "Repeat a 10,000 times and then repeat b 30,000 times" to give the string $a^{10,000}b^{30,000}$, it is not compensated intensional but it is non-compensated intensional because both parts have compression ratio much greater then 1.

Although definitions 4.2, 4.3 and 4.4 are easy to define for model-based languages which are constructed of rules, like first-order logic, equational languages, grammars…, the idea of subprogram is not so straightforward in general. In (Hernandez-Orallo 1999b) a fundamental (but neglected) question of computer science is addressed: "what is a subprogram?". Without more discussion we just present here the definitions of part and subprogram.

DEFINITION 4.5. SUBPART

The object y is a subpart of an object x in β, denoted by $y \subseteq_\beta x$, iff:

$$K_\beta(y|x) < \log K_\beta(y)$$

It is interesting to compare the definition of subpart with the notion of subset. For instance, it is easy to show that the empty string is never a subpart of any non-empty string and that most objects (but not all) are subparts of themselves. It is more intuitive to see the idea of subpart as a cognitive notion, such a subpicture.

The idea of subprogram is derived from definition 4.5:

DEFINITION 4.6. SUBPROGRAM (OR SUBTHEORY)

The object y is a subprogram of an object x in β iff

$$y \subseteq_\beta x \text{ and } \phi(y) \subseteq_\beta \phi(x)$$

DEFINITION 4.7. PROPER SUBPART

The object y is a proper subpart of an object x in β, denoted by $y \subset_\beta x$, iff $y \subseteq_\beta x$ but $x \not\subseteq_\beta y$.

DEFINITION 4.8. PROPER SUBPROGRAM (OR SUBTHEORY)

The object y is a proper subprogram of an object x in β iff

$$y \subset_\beta x \text{ and } \phi(y) \subseteq_\beta \phi(x)$$

With these later concepts, definitions 4.2, 4.3 and 4.4 are now formally settled for any descriptional mechanism or language. Finally, we can define:

DEFINITION 4.9. GREATEST EXCEPTION

With $\Delta(T)= e$ we denote the length (in bits) of the greatest exception of a description T, computed as $e = l(T) - l(G)$, where G is the smallest patched general rule of T and $e = 0$ if T has not any patched general rule.

5. An Explanatory Principle

The previous section has dealt with the distinction between intensional descriptions from non-intensional ones. However, since there are infinite many intensional descriptions (the same kind of description can be codified in infinite many ways), it would be useful to define selection criteria for intensional descriptions. In this section we present two principles or description preference criteria which avoid the possible exceptions of the MDL principle: intensional complexity and explanatory complexity.

5.1. Intensional Complexity

Straightforwardly from definition 4.8, we can define intensional complexity in the following way:

DEFINITION 5.1. INTENSIONAL COMPLEXITY

The *Intensional Complexity* (IC) of a string x on a bias β :

$$E^e_\beta(x|y) = \min \{ \ l_\beta(p_x(y)) : \Delta(p_x) \leq \ e \ \}$$

where p_x denotes any β-program for x using input y and $l_\beta(p_x)$ denotes the length of p_x in β.

In the same way as for Kolmogorov Complexity, we denote $E^e_\beta(x) = E^e_\beta(x|\ \varepsilon)$. The term $E_\beta(x) = E^0_\beta(x)$ represents the length of the shortest program for x without intrinsic exceptions.

The prior $P(h) = 2^{-E(h)}$ could be seen as an adaptation for explanation of the Occam's Razor principle ($P(h) = 2^{-K(h)}$). Under this prior, simplicity is important but secondary. Explanation is the first issue to be ensured by a description: nothing can be noise, or casual; everything is intensional, everything has a meaning, a cause. All the data must be explained.

5.2. Explanatory Complexity

However, Intensional Complexity is not still sufficient for explanation. Intuitively, something is an explanation *only if* it can be explained (and consequently related) to others.

Accordingly, there is a third factor to have in mind, *time*. There is a very appropriate way to weight space and time of a program, the formula $LT_\beta(p_x) = l(p_x) + log_2$ Cost(p_x), introduced by Levin in the seventies (see e.g. Levin 1973). A variant of $K(\cdot)$ can be easily define from it:

DEFINITION 5.2. LEVIN'S LENGTH-TIME COMPLEXITY

The *Levin Complexity* of a string x on a bias β :

$$Kt_\beta(x|y) = min \ \{ \ LT_\beta(p_x(y)) \ \}$$

This is a very practical alternative of Kolmogorov Complexity, because on behalf of avoiding intractable descriptions, it is computable. In any way, the intuitive idea of simplicity is more encompassed by a reduction of both time and space, and Occam's Razor should be better formalised by $2^{-Kt(h)}$.

Nonetheless, for our purpose of explanatory description, we must not forget intensionality. The final combination of these three factors: time, space and intensionality, gives the following definition:

DEFINITION 5.3. EXPLANATORY COMPLEXITY

The *Explanatory Complexity* (EC) of a string x on a bias β:

$$Et^e_\beta(x|y) = \min \{ LT_\beta(p_x(y)) : \Delta(p_x) \le e \}$$

In the same way as Kevin Complexity, Et^0 avoids intractable descriptions and it is computable.

5.3. The SED Principle

In the same way as the MDL principle, we can define:

Shortest Explanatory Description (SED) Principle:

"*The best model to explain a set of data is the one which minimises the sum of: the length, in bits, of the description of the theory and the data jointly; and, the logarithm of the computational cost of the description. Explicit or Intrinsic Exceptions are not allowed*".

This changes the statement that "optimal compression (*Minimum Description Length (MDL)) gives you the best hypothesis provided the data are random with respect to the hypothesis, the data are not completely perfect and the data grow to infinity*" (Vitányi and Li 1997) into the following one "the SED principle gives you a more robust hypothesis when the data are perfect". Moreover, it does not require that "the data grow to infinity", so it can be used to undertake finite real problems, where the auxiliary concepts would make it not worthy to compress.

In the framework of incremental learning, the SED criterion is less conservative than the MDL principle, and consequently it usually minimises the number of whole 'mind changes' (although these changes are usually more radical) when the data is perfect. Loosely, we should say that the MDL principle complies with Kuhn's philosophy of changing paradigms; when the number of exceptions is too great, the paradigm must be changed. In contrast, the SED usually anticipates this necessity since any exception forces the revision of the model.

One main critique to our principle is that in real problems of machine learning there is no perfect data. However, this is precisely the most practical result of *explanatory complexity*. Given some data x, if we have an expectancy of noise of about an 3%, we must only search for descriptions where $\Delta(p_x) \approx l(x) \cdot 0.03$. It is important to realise that the MDL principle gives an *uncontrollable* and *unpredictable* exception ratio, which only depends on the data and usually will underfit (for explanation) or overfit. In this way, explanatory complexity allow a less autistic evaluation of the hypotheses.

Explanatory complexity solves some of the problems (2,3,4,7) highlighted in section 3. Other problems (1, 5, 8) are softened. However, the last one can still be solved completely and it is the most relevant one for scientific discovery and, as we will see, for a complete re-understanding of what is to learn.

6. Computational Information Gain

In example 4.1, we can observe that the n−1 order polynomial for n points of data (description D_5), although intensional, has few informative value. The question is to distinguish which descriptions are really valuable or, in a relative way, which objects are valuable wrt. other objects. This can be particularised for studying which hypotheses are valuable wrt. the data.

DEFINITION 6.1. COMPUTATIONAL INFORMATION GAIN

The *Computational Information Gain* of an object x wrt. an object y:

$$G(x \mid y) = Kt(x \mid y) \,/\, Kt(x)$$

The rationale of the definition is to measure to which extent the use of y is useful for describing x.

THEOREM 6.2. Limits of $G_\beta(x \mid y)$

For every x and y, $log\ l(x)/(l(x) + log\ l(x)) \overset{+}{<} G(x \mid y) \leq 1$.

PROOF. The second inequality $G(x \mid y) \leq 1$ is obvious by choosing $y = \varepsilon$ and the definition of $Kt(x)$ as $Kt(x \mid \varepsilon)$. The first inequality is justified by the fact that the numerator

$$Kt(x \mid y) \geq \log l(x) \qquad (1)$$

because x must be printed and this takes at least $l(x)$ units of time. In fact this limit can be come close if $x = y$ because the program "print y" has cost approximately $2 \cdot l(x)$ for reading and writing x.

The denominator must follow the relation

$$Kt(x) \overset{+}{<} l(x) + log\ l(x) \qquad (2)$$

because in the worst case, when x is random, we need $l(x) + c$ bits of information for the program "print x" and at least $l(x)$ units of time to be printed.

From (1) and (2) we have that $log\ l(x)/(l(x) + log\ l(x)) \overset{+}{<} G(x \mid y)$. □

Before to interpret definition 6.1 in the following section, it is interesting to study if this definition could be 'cheated' in some way. For instance, if we have an efficient method to go from y to x, then, intuitively, it is not much valuable to have y than to have x, because we can easily go from y to x. The following theorem states that this intuition is captured well by definition 6.1.

THEOREM 6.3. Robustness to polynomiality

Consider a *learning* or *discoverer* algorithm A^* in $\mathbb{P}$ (i.e. polynomial), namely $\exists p \in \mathbb{N} : O(n^{p-1}) \leq O(A^*) \leq O(n^p)$, being A^* of constant size, i.e., $l(A^*) = c$, such that this algorithm deterministically transforms y into x, where x is a program for y, being $n = l(y)$.

$$\text{If } Kt(x) > k \cdot p \cdot \log n, \text{ then } G(x \mid y) \leq 1 / k.$$

In other words, if x is complex, but it can be easily obtained from y, then $G(x|y)$ is low.

PROOF. For every string of data y, let us construct x in the following way: x = "apply A^* to y". Since we can construct x from $\langle A^*, y\rangle$ in an easy way p = "apply 1st argument to 2nd argument" $Kt(x \mid \langle A^*, y\rangle) \leq LT(p) = l(p) + \log cost(p) \leq c + \log n^p$). It is obvious that $Kt(x \mid y) \overset{+}{\leq} Kt(x \mid \langle A^*, y\rangle)$. So we have that $Kt(x \mid y) \leq \log n^p = p \log n$.

If, as supposed, $Kt(x) > k \cdot p \cdot \log n$, then the quotient $G(x|y) = Kt(x|y) / Kt(x) \leq 1 / k$. □

7. Informativeness and Creativity

Most selection criteria (as the MDL or our SED principle) talk about "the best model". However, this is a fallacy, because if we remove all the models which are long, intractable or non-explanatory there may be still many good models for a given data. In this case, it is only reasonable to talk about "the best model" if this model is significantly better than the second best model. But after all, the second best model is usually the first model with a slight modification. In other words, there are no discontinuities in the goodness of models. Thus, it would be more percipient to talk about an absolute goodness of a model, how short it is, how intensional it is, how explanatory, etc.

But it would be even more insightful to evaluate how much valuable is to obtain a concrete description and whether it is worthy to remember or forget it. This would be especially useful if an inductive method can consider different hypotheses at a time, because some surprising, strange, difficult to obtain, or curious hypotheses which have not been still refuted can be kept for future use. On the other hand, obvious or easy hypotheses can be forgotten because they would be easily generated again when needed.

In this way, $G(x \mid y)$ provides a uniform measure of the relative value of the hypothesis wrt. the data, the gain of the computational effort which has been invested in the process from the data to the hypothesis. More precisely, if x is the theory and y is the data, the two extreme cases are illustrative:

- Minimum: $G(x \mid y) = \log l(x) / (l(x) + \log(l(x)) \approx 0$. The theory is *evident* from the data. It is very easy to describe the theory from the data. Some examples which can produce this minimum are: the polynomial obtained using the data, a description full of exceptions or full of great extensionalities because they can be described easily from the data.
- Maximum: $G(x \mid y) = 1$. The theory is *surprising* or *creative* wrt. the data. The data is useless (in time-space terms) to describe the theory ($Kt(x \mid y) = Kt(x)$). It is necessary a great computational work on the data y to obtain the theory or

there is a need for external information. In other words, the computational effort invested justifies x to be retained.

7.1. Informative Hypotheses

This engages with the classical dilemma between informative and probable hypotheses. It is clear that an explanation must have some degree of plausibility to avoid fantastic hypotheses, but in many applications, like scientific discovery or abduction, we must regard an explanation as an investment, even a "risky bet" that could be soon falsified. This is merely Popper's criterion of falsifiability (Popper 1962): one does not always want the most likely explanation, because sometimes it is the less informative too.

The issue is clear when the data are random (and this usually happens with short data because it makes no worthy any compression). The MDL principle just gives the data themselves, which does not correspond to the idea of 'model'. By forcing a gain near to 1, different informative hypotheses can be induced. This gives clues to the enigma of "hyper-learning" or "poverty of stimulus" in those cases where the data suggests some obvious (but useless) hypothesis instead of more creative ones.

Moreover, deduction can be informative, something that Hintikka vigorous vindicated (Hintikka 1970), which places deduction and induction as either informative or non-informative processes depending on $G(x \mid y)$, y being the data and x being the inferenced result (an inductive hypothesis or a deductive derivation). This is contrary to the traditional idea of induction as always information increasing inference process and deduction as always information decreasing inference process (Bar-Hillel and Carnap 1953) and provides ways to solve problem 6 of section 3.

7.2. What is to Discover? What is to Learn?

A further insight in the learning of finite data indicates that if the hypothesis is evident from the data, no much learning has taken place. However, the most important learning paradigms are based on the idea of identification: identification in the limit (Gold 1967), PAC model (Valiant 1984), Query-Learning (Angluin 1988). However, these paradigms are designed for infinite data, because a learning algorithm that always gives a complete extensional (and not valuable) description "print x" for any finite data x would formally learn, something that is quite counterintuitive.

We can say that a concept or theory x is an *authentic learning* or *discovering* wrt. y in a context β iff $G_\beta(x \mid y)$ is close to 1 and $G_\beta(y \mid x)$ is close to 0, i.e, x is surprising for y and x is an efficient theory for y. In a proper way, discovering should be accompanied by a confirmation, whereas learning must not necessarily be confirmed, because x is valuable *per se*.

From here, and very far from the classical notion of 'identification', we propose a different notion of learning (or discovering): the more a system learns the more valuable the description is with respect to the data.

8. Relation between Explanation and Informativeness

Computational Information Gain was motivated by the fact that intensional does not mean creative or informative. The last descriptions of example 4.1 are easy to obtain from the data, so they are not much valuable. Moreover, the construction of the $n-1$ order polynomial for n points of data is a systematic method, so there is always an 'easy' intensional description for any evidence. The major coincidence between intensionality and a high value of $G(x|y)$ is that extensional quoting are avoided, as the next theorem shows:

THEOREM 8.1

Given an efficient description x for a long data y, such that x contains a sequential quoting Q of a random sequence q from y of reasonable size, namely, $l(q)=e > \log^2 l(y)$, then x is not intensional and $G(x|y) < 1-e/l(x)$.

For instance, 1,000 bits of data with a description of lengh 200 bits that contains a sequential quoting of 120 bits is intensional and $G(x|y) < 0.4$.

PROOF. Since Q is a quoting like "Print y_k, y_{k+1}, ..., y_{k+e-1}" then $CR(Q) = e \ / \ \{mf_q \cdot e + af_q\} \leq 1$. The first assertion, x is not intensional, is obvious by choosing G as the rest of T removing Q.

Since $n > 1$, the compression of the whole theory $CR(T) > 1$, then $CR_{\phi(T)}(G) \geq CR(T)$ because $CR(Q) \leq 1$, and $l(\phi(T) - \phi(G)) \ / \ \{mf_q \cdot (l(T) - l(G)) + af_q \} \leq 1$, because the first term is precisely $CR(Q)$.

The second assertion, $G_\beta(x \mid y) = Kt(x \mid y) \ / \ Kt(x)$. Since there is a part of x which is exactly in y, it can be recognised from the input y by only selecting the beginning of the sequence in y and the length e. Coding this information $Kt(q \mid y)$, in any case, cannot be greater in length than $\log(l(y)) + c_l$, because a position can be coded by a usual digital notation and it cannot be greater in time than $l(y) + c_t$, to traverse the sequence y. Jointly, we have that $Kt(q \mid y) \leq \log(l(y)) + c_l + \log(l(y) + c_t) = 2 \cdot \log(l(y)) + c_{lt}$. Since y is long, c_{lt} can be ignored.

Since q is random, $Kt(q) \geq l(q) + \log l(q) = e + \log e \geq e$. The term $Kt(x)$ can be decomposed into the cost of describing q and the code of describing the rest, say g, namely, $Kt(x) = Kt(g) + Kt(q)$. However, $Kt(x \mid y)$ is exactly $Kt(g \mid y) + Kt(q \mid y)$. Since $Kt(g \mid y)$ is always less or equal than $Kt(g)$ and we have stated that $Kt(q \mid y) \leq 2 \cdot \log(l(y))$ then $Kt(x \mid y) \leq Kt(g) + 2 \cdot \log(l(y))$. From here, $G(x \mid y) = Kt(x \mid y) \ / \ Kt(x) \leq \{Kt(g) + 2 \cdot \log(l(y))\} \ / \ \{Kt(g) + Kt(q) \ \} \leq$

$\{Kt(g) + 2 \cdot \log(l(y))\ \} / \{\ Kt(g) + e\ \} = \{Kt(g) + 2 \cdot \log(l(y)) + e - e\ \} / \{\ Kt(g) + e\ \} = 1 - \{\ e - 2 \cdot \log(l(y))\ \} / \{\ Kt(g) + e\ \}$. Since $e > \log^2(l(y))$ and l(y) is long we can ignore the term $\log(l(y))$, giving $G(x \mid y) \leq 1 - e\ / \{\ Kt(g) + e\ \}$

Since $Kt(g) + Kt(q) = Kt(x)$, by using again the value of $Kt(q)$, then we have that $Kt(g) \leq Kt(x) - e$ and we finally have that $G(x \mid y) \leq\ 1 - e / \{\ Kt(x) - e + e\ \} = 1 -\ e / Kt(x)$ and since $\log l(x) \leq Kt(\mathrm{x}) \leq l(x) + \log l(x) \approx l(x)$ then $G(x \mid y) \leq\ 1 -\ e / l\ (x)$. □

Apart from these commonalties, they express quite different but compatible notions which are worthy to combine. The idea is to obtain explanatory descriptions and to preserve those which are valuable in terms of computation gain. In other words, free computational resources (time and space) should be invested in informative hypotheses.

9. Conclusions

In this paper we have critically discussed the maxim of "learning as compression". In any case, the two major problems of MDL's 'autism': explanation and informativeness make that the maxims "explanation as compression" and "discovering as compression" are not sustainable.

We have introduced two different solutions for both problems. First, we have presented "Explanatory Complexity" to address the problems of Kolmogorov Complexity for explanation. Secondly, we have elaborated the idea of "Computational Information Gain" to clarify what is an informative hypothesis and to give more light to the blurry notions of surprise, discovering and creativity. In the end, computation information gain can be used to give a more reliable certification that real learning has take place.

As a result we are able to counter two assertions from the advocators of the MDL principle. Their first claim is: "*a model that is much too complex is worthless, while a model that is much too simple can still be useful*." (Grünwald, 1999). Our response is that a model that is evident or extensional is worthless, while a surprising model or intensional can still be useful. In the same line, Grünwald presents "*another way of looking at Occam's Razor*" as "*If your overfit, you think you know a lot but you do not. If you underfit, you do not know much but you know that you do not know much. In this sense, underfitting is relatively harmless while overfitting is dangerous*". However, since *most* of data sequences are incompressible, the MDL principle gives no knowledge at all, in *general*. Maybe not knowing, i.e., ignorance, is relative harmless, but it is also useless.

In conclusion, the MDL principle works well in those environments where the bias does not allow extensional descriptions or where the data are huge and from statistical or imperfect sources. But, when faced to a concrete learning problem or in scientific discovery, we have to tune length, computational time, intensionality and informativeness of descriptions according to the expectation we have about the source of knowledge. In our view, Occam's Razor should be understood in this non-autistic way.

References

Angluin, D., 1988, Queries and concept learning, *Machine Learning* 2, 4:319-342.

Barker, S.F., 1957, *Induction and Hypothesis*, Ithaca.

Bar-Hillel, Y. and Carnap, R., 1953, Semantic information, *British J. for the Philosophy of Science* 4:147-157.

Blum, M., 1967, A machine-independent theory of the complexity of recursive functions, *J. ACM* 14, 4:322-6.

Blum, L. and Blum, M., 1975, Towards a mathematical theory of inductive inference. *Inform. and Control* 28:125-155.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M., 1989, Learnability and the Vapnik-Chervonenkis dimension, *Journal of ACM*, 36:929-965.

Board, R., and Pitt, L., 1990, On the necessity of Occam algorithms, in Proc., 22nd ACM Symp. Theory of Comp..

Bosch, van den, 1994. *Simplicity and Prediction*, Master Thesis, dep. of Science, Logic & Epistemology of the Faculty of Philosophy at the Univ. of Groningen.

Case J. and Smith, C., 1983, Comparison of identification criteria for machine inductive inference, *Theoret. Comput. Sci.* 25:193-220.

Cheeseman, P., 1990, On finding the most probable model, in: Shrager, J. and Langley, P. *Computational models of scientific discovery and theory formation*, Morgan Kaufmann.

Conklin, D. and Witten, I. H., 1994, Complexity-based induction, Machine Learning, 16:203-225.

Derthick, M., 1990, The minimum description length principle applied to feature learning and analogical mapping, MCC Tech. Rep. no. ACT-CYC-234-90.

Ernis, R., 1968, Enumerative induction and best explanation, *J. Philosophy*, LXV (18): 523-529.

Freivalds, R., Kinber, E., Smith, C.H., 1995, On the intrinsic complexity of learning, *Information and Control* 123:64-71.

Gold, E.M. 1967, Language identification in the limit, *Information & Control*,10:447-474.

Grünwald, P., 1999, Model selection based on minimum description length, submitted to *Journal of Mathematical Psychology*. CWI, Amsterdam.

Gull, S.F., 1988, Bayesian inductive inference and maximum entropy, In: *Maximum Entropy and Bayesian Methods in Science and Engineering*, Vol. 1: Foundations, ed. by G.J. Erickson and C.R. Smith, pp. 53-74. Dordrecht: Kluwer.

Harman, G., 1965, The inference to the best explanation, *Philos. Review*, 74: 88-95.

Hempel, C.G., 1965, *Aspects of Scientific Explanation*, The Free Press, New York.

Hernandez-Orallo, J., 1999a, Constructive reinforcement learning, *International Journal of Intelligent Systems*, to appear.

Hernandez-Orallo, J., 1999b, What is a subprogram?, *submitted*.

Hernandez-Orallo, J. and Garcia-Varea, I., 1998, Distinguishing abduction and induction under intensional complexity, in: *Proceedings of the ECAI'98 Workshop on Abduction and Induction* in AI, Flach, P.; Kakas, A., eds., 41-48, Brighton.

Hintikka, J., 1970, Surface information and depth information, in: *Information and Inference*, Hintikka, J.; Suppes, P., eds., D. Reidel Publishing Company, 263-297.

Kearns, M., Mansour, Y., Ng, A.Y., and Ron, D., 1999, An experimental and theoretical comparison of model selection methods, *Machine Learning*, to appear.

Kuhn, T.S., 1970, *The Structure of Scientific Revolutions*, University of Chigago.

Levin, L.A., 1973, Universal search problems, *Problems Inform. Transmission*, 9:265-266.

Li, M.; Vitanyi, P., 1997, *An Introduction to Kolmogorov Complexity and its Applications*, 2nd Ed. Springer-Verlag.

Muggleton, S., Srinivasan, A., and Bain, M., 1992, Compression, significance and accuracy, in: D. Sleeman and P. Edwards (eds.) Machine Learning: Proc. of the 9th Intl Conf (ML92), 523-527, Wiley.

Muggleton, S. and De Raedt L., 1994, Inductive logic programming — theory and methods, *J. of Logic Prog.*, 19-20:629-679.

Pfahringer, B., 1994, Controlling constructive induction in CiPF: an MDL approach, in F. Bergadano and L. de Raedt (eds) Machine Learning, *Proc. of the European Conf. on Machine Learning* (ECML-94), pp. 242-256, LN AI 784, Springer-Verlag 1994.

Popper, K.R., 1962, *Conjectures and Refutations: The Growth of Scientific Knowledge*, Basic Books, New York.

Quinlan, J.; Rivest. R., 1989, Inferring decision trees using the minimum description length principle, *Information and Computation*, vol. 80, pp. 227-248.

Rissanen, J., 1978, Modeling by the shortest data description, *Automatica-J.IFAC*, 14:465-471.

Rissanen, J., 1986, Stochastic complexity and modeling, *Annals Statist.* 14:1080-1100.

Rissanen, J., 1996, Fisher information and stochastic complexity, *IEEE Trans. on Information Theory*, 42(1).

Rivest, R.L., and Sloan, R., 1994, A formal model of hierarchical concept learning, *Inf. and Comp*. 114:88-114.

Schaffer, C., 1994, A conservation law for generalization performance, in Proc. of the 11th Intl. Conf. on Machine Learning, 259-265, 1994.

Sharger, J. and Langley, P., 1990, *Computational Models of Scientific Discovery and Theory Formation*, Morgan Kaufmman, 1990.

Solomonoff, R.J., 1964, A formal theory of inductive inference, *Inf. Control.*, 7:1-22, Mar., 224-254, June.

Solomonoff, R.J., 1978, Complexity-based induction systems: comparisons and convergence theorems, *IEEE Trans. Inform. Theory*, IT-24:422-432.

Valiant, L., 1984, A theory of the learnable, Comm. of the ACM 27(11): 1134-1142.

Vitányi, P., and Li, M., 1996, Minimum description length induction, bayesianism, and Kolmogorov complexity.

Vitányi, P., and Li, M., 1997, On prediction by data compression, in: Proc. of the 9th European Conf. on Machine Learning, LNAI 1224, Springer-Verlag, 14-30.

Wallace, C.S., and Boulton, D.M., 1968, An information measure for classification, *Computing Journal* 11:185-195.

Watanabe, S., 1972, Pattern recognition as information compression, in: Watanabe (ed.) *Frontiers of Pattern Recognition*, New York: Academic Press.

Wolff, J.G., 1995, Computing as compression: an ovverview of the SP theory and system, *New Gen. Computing* 13:187-214.

Wolpert, D., 1992, On the connection between in-sample testing and generalization error, *Complex Systems*, 6:47-94.

Zemel, R., 1993 *A minimum description length framework for unsupervised learning*. Ph. D. Thesis, Dept. of Computer Science, Univ. of Toronto.